



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/773,121	01/31/2001	John C. Adams	10006660-1	5388

7590                    06/04/2004

HEWLETT-PACKARD COMPANY  
Intellectual Property Administration  
P.O. Box 272400  
Fort Collins, CO 80527-2400

EXAMINER

STEELMAN, MARY J

ART UNIT

PAPER NUMBER

2122

DATE MAILED: 06/04/2004

4

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/773,121	ADAMS, JOHN C. <i>[Signature]</i>	
	<b>Examiner</b>	<b>Art Unit</b>	2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

#### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

1) Responsive to communication(s) filed on 18 March 2004.

2a) This action is FINAL.                    2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

4) Claim(s) 21-39 is/are pending in the application.

4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.

5) Claim(s) \_\_\_\_\_ is/are allowed.

6) Claim(s) 21-39 is/are rejected.

7) Claim(s) \_\_\_\_\_ is/are objected to.

8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a) All    b) Some \* c) None of:

1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413)
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	Paper No(s)/Mail Date. _____.
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
	6) <input type="checkbox"/> Other: _____.

## **DETAILED ACTION**

1. This office action is in response to Amendment A, received 18 March 2004. Per Applicant's request, claims 1-20 are canceled. New claims 21-39 are pending.

### ***Claim Rejections - 35 USC § 112***

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 23 and 35 recite the limitation "...from the group...". There is insufficient antecedent basis for this limitation in the claim. Claims should recite 'from a group'. Delete 'the' and insert 'a'.

4. In view of the cancellation of all prior claims, the 35 USC 112 rejections of the former Office Action are hereby withdrawn.

### ***Claim Objections***

5. Claim 32, first line, recites, "A method for providing executable software program that automatically generating...", should be --A method for providing executable software program for automatically generating....-- Delete 'that' and insert 'for'.

### ***Claim Rejections - 35 USC § 103***

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 21, 22, 24, 26, 28-34, 36, and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent 6,035,119 to Massena et al., in view of US Patent 6,598,035 B2 to Branson et al.

Per claim 21:

Massena disclosed (Abstract, lines 1-11), "...a method, apparatus, and medium (computer readable medium) for adding text and text-based components to a Web page hosted on a server. A control, which is run at the designing time of the web page...writes HTML information to a created web page. The created HTML information may include text and other text based components (client and server scripting, applets, ActiveX controls, JAVA scripting, and other components). Through the use of OLE, the controls incorporate author-friendly capabilities including in-place editing, property sheets, and persistence..." Thus Massena disclosed, "software that programmatically generates a markup language document having at least one markup language element", as set forth in the preamble of claim 21. Also see col. 2, lines 47-49, "...automatically generates HTML text and script onto a web page."

-a plurality of code sections each automatically adding to the markup language document a corresponding markup language element,

(Col. 3, lines 11-16 and 31-36, "...the present invention uses a Component Object Model (COM) component...The controls may be...ActiveX controls that automatically generate HTML text..." and "Design-time controls are similar to embedded wizards that can be continuously edited to

modify the text they generate for a web page. DTCs take advantage of OLE design-time capabilities (including direct manipulations, property sheets, and persistence)..." Also col. 4, lines 53-57, "The developer may insert a design-time control into the file to provide a desired functionality to the web page. The design-time control may be any control capable of generating text and/or computer-executable code.")

wherein each code section utilizes one of the following object-oriented classes each of which models a type of markup language element:

(Col. 2, lines 65-67, "...the present invention may be used to generate C++ code in a C++ development environment. (uses object oriented classes)")

Although Massena disclosed creating an HTML document, he failed to discuss the defined features of markup language known in the art and related design decisions. HTML is derived from SGML and additional well known details may be obtained from the w3.org consortium. (<http://www.w3.org/TR/WD-xml-961114.html>):

" Extensible Markup Language, abbreviated XML, describes a class of data objects stored on computers and partially describes the behavior of programs which process these objects. Such objects are called XML documents. XML is an application profile or restricted form of SGML, the Standard Generalized Markup Language [ISO 8879].

XML documents are made up of storage units called entities, which contain either text or binary data. Text is made up of characters, some of

which form the character data in the document, and some of which form markup. Markup encodes a description of the document's storage layout, structure, and arbitrary attribute-value pairs associated with that structure. XML provides a mechanism to impose constraints on the storage layout and logical structure.

A software module called an *XML processor* is used to read XML documents and provide access to their content and structure. It is assumed that an XML processor is doing its work on behalf of another module, referred to as the *application*. This specification describes some of the required behavior of an XML processor in terms of the manner it must read XML data, and the information it must provide to the application.

### 1.1 Origin and Goals

XML was developed by a Generic SGML Editorial Review Board formed under the auspices of the W3 Consortium in 1996 and chaired by Jon Bosak of Sun Microsystems, with the very active participation of a Generic SGML Working Group also organized by the W3C. The membership of these groups is given in an appendix."

While Massena provided information on creating HTML documents, he failed to disclose extensible objects, and rules and conditions attached to the objects to create specific behavior. However, Branson disclosed:

-inline class that models a first type of markup language element in which other markup language elements cannot be nested, and which comprises an opening tag, an argument and no closing tag;

(Branson: Col. 2, lines 14-24 & 58-63, "...a reusable object oriented (OO) framework for use with object oriented programming systems comprises an expert system shell that applies a rule set developed by a framework user against a user-defined set of conditions ...thereby provides a more readily developed rule-based system. The framework includes a condition manager object that keeps track of what conditions are true at any time. The conditions are a set of object oriented classes that are organized into an inheritance hierarchy for use by the condition manager." And "The condition manager and rules compiler comprise a condition management facility that are provided as core classes...The set of rules and conditions comprise extensible object oriented classes that are customized by the expert system developer." Rules and conditions are a design decision. A class can be derived, named and defined with rules regarding nesting, tags, and arguments. )

-a container class that models a second type of markup language element in which other markup language elements can be nested, and which comprises an opening tag, a closing tag and an argument disposed between the opening and closing tags.

(Multiple classes exist and model mark up language elements. Branson disclosed a 'container' at col. 7, line 31, "...a class that defines an object that is a container for all zoo keeper objects...",

Art Unit: 2122

as an example of a container ‘object’ that holds other objects. Nesting, tags, and enclosed arguments are addressed as rules of the classes. The ability to nest objects, include tags, and enclose arguments is well known in mark up language. Rules can be made regarding nesting, tags, and arguments.)

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Massena’s invention of producing mark up language documents by including well known defined features of HTML such as objects, classes, derived classes, and tags. And furthermore to include permissive and restrictive rules and conditions, as supplied by Branson when creating and designing classes and relationships because these are all known abilities in the mark up language as originally defined by W3 Consortium.

Per claim 22:

-the object-oriented classes further comprise a base class defining a parent-child relationship by which a child object is stored within storage space of the child’s parent object;

(Branson disclosed (col. 15, lines 23-29), “FIG. 11 is a category diagram for the framework implemented in the computer system of FIG. 8...the categories illustrated in FIG. 11 correspond to object oriented programming (OOP) objects that encapsulate data attributes and behaviors (OOP is defined by objects encapsulating behavior and data, inheritance, parent / child relationships, extensibilities...), and are stored in the memory 38 (child object is stored within storage space of child’s parent object) illustrated in the block diagram of FIG. 8.” )

-wherein the inline class is an extension of the base class, wherein each object of the inline class is permitted to be a child object and prohibited from being a parent object;

(Branson disclosed that some objects would be prohibited from being a parent object: (col. 4, lines 61-64) "...part of the framework that is not to be subject to potential customization and extension (cannot be a parent)..." Also, Col. 16, lines 36-41, "...the framework user is provided only with one core category. All of the other mechanisms, the Detector, Rules, and Conditions, are indicated as extensible...framework user is free to customize the objects (derive classes and create rules and conditions) of these mechanisms." The framework can be extended and customized. Child / parent relationships can be defined and restricted using the rules and conditions.)

-wherein the container class is an extension of the base class, wherein each object of the container class is permitted to be either a child object or a parent object;

(Branson: Col. 16, lines 36-41, "...the framework user is provided only with one core category. All of the other mechanisms, the Detector, Rules, and Conditions, are indicated as extensible...framework user is free to customize the objects (derive classes and create rules and conditions) of these mechanisms." The framework can be extended and customized. Child / parent relationships can be defined and restricted using the rules and conditions.)

Massena disclosed “markup language element modeled by an object” at col. 3, lines 31-36, “Design-time controls...can be continuously edited to modify the text they generate...take advantage of OLE design-time capabilities...thereby extending the capabilities...” and col. 4, lines 53-57, “The developer may insert a design-time control into the file to provide a desired functionality to the web page...” The control is an object that models markup language.

Massena failed to disclose the remaining part of the limitation “depends on the parent-child relationship defined for the class which the object is a member” However, Branson disclosed a framework for use with object-oriented programming systems. A rule set and conditions are used to indicate an appropriate action. (Abstract, lines 1-17.) Rules and conditions enforce the parent child relationship as defined in the class.

Therefore, it would have been obvious, to one of ordinary skill in the art, to modify Massena’s invention that generates HTML documents, to include rule and condition setting features when deriving and creating various classes, as disclosed by Branson in an object-oriented environment because (Branson, col. 1, lines 16-17) “An expert system emulates human reasoning within a collected knowledge base.” (Branson, col. 1, lines 60-61): “...many expert systems are tailored for each particular application...” (Branson, col. 2 lines 8-11): “...apparent that there is a need for an expert system development mechanism tool that provides a basis for more rapid and easy development o rule-based expert systems.” The object oriented features are well known in the art.

Art Unit: 2122

Per claim 24:

-a least one class that is an extension of the inline class and which, when processed by a browser, produces a corresponding section of a web page comprising a first type of document item.

(Massena disclosed (col. 5, lines 48-54), “Controls are capable of handling their own drawing...providing content menus and other functions (providing types of document items) which generally extend (class extended) the graphical editing environment of a container. In addition, controls can use a property browser and property page frames of their container to allow the user to set properties of the control.”)

Per claim 26:

-at least one class that is an extension of the container class and which, when processed by a browser, produces a corresponding section of a web page comprising a second type of document item.

(Massena disclosed (col. 5, lines 45-46, “Controls may, for example, when embedded within a we page, present a graphical user interface (extension of a container class produced as a corresponding section of a web page) within the area of the control (where located on the web page) when displayed on a browser.”)

Per claim 28:

Art Unit: 2122

-software includes a variable declaration section in which is declared a variable defining a data structure that ultimately contains the markup language document.

(Massena disclosed (col. 2, lines 51-55), "When the file is saved, the editor or container asks the control for both its design-time information (variable definition) as well as its run-time HTML text and run-time script which should be written into the file. The design-time information of the control is then rendered invisible...")

Per claims 29, 30, and 31:

-the members of the container class each utilize temporary files to store its information.  
-a class that is an extension of the inline class and which utilizes a buffer data structure.  
-child objects of the inline class and the container class can use their parent's storage when it is available.

(It is common knowledge that classes, extended classes and objects instantiated from those classes may store information in a temporary file, a buffer data structure, or in memory of many well known types and thus is obvious. Massena: col. 6, lines 31-44, "...environment described herein employs a hard disk, a removable magnetic disk, and a removable optical disk, it should be appreciated by those skilled in the art that other types of computer readable media which can store data that is accessible by a computer...may also be used in the exemplary operating environment. A number of program modules may be stored on the hard disk, magnetic disk, optical disk, ROM, or RAM, including...other program modules and program data...")

Per claim 32:

Massena disclosed (Abstract, lines 1-11), "...a method, apparatus, and medium for adding text and text-based components to a Web page hosted on a server. A control, which is run at the designing time of the web page...writes HTML information to a created web page. The created HTML information may include text and other text based components (client and server scripting, applets, ActiveX controls, JAVA scripting, and other components). Through the use of OLE, the controls incorporate author-friendly capabilities including in-place editing, property sheets, and persistence..."

-developing a program source code on a computer readable medium comprising a plurality of code sections each automatically adding to the markup language document a corresponding markup language element, wherein the markup language elements comprise:

(See limitations as addressed in claim 1 above.)

a first type of markup language element in which other markup language elements cannot be nested and which comprises an opening tag, an argument and no closing tag;

(See limitations as addressed in claim 1 above.)

-a second type of markup language element in which other markup language elements can be nested and which comprises an opening tag, a closing tag and an argument disposed between the opening and closing tags;

Art Unit: 2122

(See limitations as addressed in claim 1 above.)

-linking the source code to a library in which is packaged source code of the object-oriented classes each of which models either the first type or the second type of markup language element;

(Massena: col. 7, lines 41-43, “One embodiment of the invention includes third parties hosting design-time controls which are downloaded as needed to an author’s editing environment.”

Download various desired controls and ‘link’ into editing environment. Also, col. 7, lines 55-60, “The design-time control is then ready to be compiled...the control may be installed...in the computer system’s registry...” and col. 8, lines 13-15, “HandleComponentRegistration – registers the controls in this library as design-time controls...” The controls generate HTML text and script.)

-compiling the linked code to produce the executable software program that, when executed, processes a first set of input conditions to automatically generate the markup language document.

(Massena: col. 7, lines 55-60, “The design-time control is then ready to be compiled...the control may be installed...in the computer system’s registry...” and col. 8, lines 13-15, “HandleComponentRegistration – registers the controls in this library as design-time controls...” Link and compile the controls to be used (col. 7, lines 37-38) “to author any text-based solution including HTML tags and content...”)

Per claim 33:

-providing a set of classes each of which models a type of markup language element comprising:

-an inline class that models the first type of markup language element;

(See limitations as addressed in claim 1 above.)

-a container class that models the second type of markup language element;

(See limitations as addressed in claim 1 above.)

-declaring one or more variables of the types defined by the classes or extensions thereof;

(Massena: col. 4, lines 53-57, “The developer may insert a design-time control into the file to provide a desired functionality to the web page. The design-time control may be any control capable of generating text ...” To insert a control is to declare a variable of the control object type.)

-calling any functions defined by the classes and extensions thereof;

(Massena: Col. 5, lines 20-22, ...controls which can be inserted into web pages to provide desired functionality...”, col. 5, lines 30-32, “Through the execution of the design-time control during...the saving process (save function called by class), the run-time text is created and the design-time control commented out.”)

-coding other parts of the program source code as necessary.

(Massena: Col. 5, lines 48-54, "Controls are capable of handling their own drawing, accepting mouse and keyboard inputs, providing content menus and other functions which generally extend the graphical editing environment of a container. In addition, controls can use a property browser and property page frames of their container to allow the user to set properties of the control." Controls can be extended (coded) to handle drawing, inputs, provide menus and add properties to the content.)

Per claim 34:

-a base class defining a parent-child relationship by which a child object is stored within the storage space of the child's parent object;

(See limitations addressed in claim 22 above.)

-wherein the inline class is an extension of the base class, wherein each object of the inline class is permitted to be a child object and prohibited from being a parent object;

(See limitations addressed in claim 22 above.)

-wherein the container class is an extension of the base class, wherein each object of the container class is permitted to be either a child object or a parent object;

(See limitations addressed in claim 22 above.)

Art Unit: 2122

-wherein the type of markup language element modeled by an object depends on the parent-child relationship defined for the class of which the object is a member.

(See limitations addressed in claim 22 above.)

Per claim 35:

-markup language is selected from the group consisting of HTML; XML; XHTML; and SGML.

(See limitations as addressed in claim 23 above.)

Per claim 36:

-at least one class that is an extension of the inline class and which, when processed by a browser, produces a corresponding section of a web page comprising a first document item.

(See limitations as addressed in claim 24 above.)

Per claim 38:

-at least one class that is an extension of the container class and which, when processed by a browser, produces a corresponding section of a web page comprising a second document item.

(See limitations as addressed in claim 26 above.)

8. Claims 23 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent 6,035,119 to Massena et al., in view of US Patent 6,598,035 B2 to Branson et al., and further in view of US Patent 6654754 to Knauft et al.

Massena disclosed (Abstract, lines 1-11), "...a method, apparatus, and medium (computer readable medium) for adding text and text-based components to a Web page hosted on a server. Massena disclosed "software that programmatically generates a markup language document having at least one markup language element". Also see col. 2, lines 47-49, "...automatically generates HTML text and script onto a web page."

Branson disclosed an object oriented rule-based expert system framework. A rule set determines classes and inheritance policies. Branson failed to specify that this framework could be used in generating documents using various markup languages. Branson failed to disclose, "-markup language is selected from the group consisting of HTML; XML; XHTML; AND SGML"

However, Knauft disclosed dynamically generating documents (col. 6, lines 24-28). See fig. 2, #216A-216N. Knauft disclosed, at col. 8, lines 23-28, "The data objects 216A-216N can be of any arbitrary format and can contain any type of data. For example the data objects 216A-216N can include: an electronic document according to any open or proprietary format, i.e., HTML...SGML...XHTML, XML..."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Massena / Branson to include many types of mark up documents, such as HTML, XML, XHTML and SGML documents as noted in Knauft's patent, because they are well known in the art, and when generating documents, using an object oriented framework, the mark up languages permit documents to be widely used over a network. HTML, XML AND XHTML are versions of the original SGML, Standard Generalized Markup Language.

Art Unit: 2122

9. Claims 25, 27, 37 and 39 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent 6,035,119 to Massena et al., in view of US Patent 6,598,035 B2 to Branson et al., and further in view of US Patent 5,708,825 to Sotomayor.

Per claims 25, 27, 37 and 39:

-the first type document item / second type document is selected from a group consisting of comment text; formatted text; embedded text; and image; an anchor; a paragraph marker; a line break; and a horizontal rule / bold text item; a horizontally centered item; a table; a subdocument; and a selection list.

Massena disclosed (col. 5, lines 48-54), “Controls are capable of handling their own drawing, accepting mouse and keyboard inputs, providing content menus and other functions (providing types of document items) which generally extend (class extended) the graphical editing environment of a container. In addition, controls can use a property browser and property page frames of their container to allow the user to set properties of the control.” Also, col. 7, lines 36-37, “One may use design-time controls to author any text-based solution including HTML tags and content...ActiveX controls...and other text-based solutions...” Also Col. 2, lines 54-58, “The design-time information of the control is then rendered invisible to the run-time user by being wrapped inside an HTML comment (comment text). The control is thereby made invisible...” Branson disclosed an object oriented rule-based expert system framework. A rule set determines classes and inheritance policies. The combination failed to specify an extended class, inline or container, corresponds to a document item selected from a

group consisting of comment text, formatted text, embedded text, an image, an anchor, a paragraph marker, a line break and a horizontal rule / bold text item, horizontally centered item, table, subdocument, and a selection list.

However Sotomayor disclosed scanning documents to generate presentation pages (Abstract, line 1-2 & 13). Tags, enclosing defining tokens, are well known in the art and are used for formatting, comments, embedding, etc. As an example, Sotomayor disclosed comment tokens at col. 27, lines 61-64, horizontal rules at col. 30, lines 30 and 54, align middle col. 30, line 37, anchors at col. 31, line 22, paragraph marks at col. 30, lines 4-6, new line at col. 33, line 59.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention to modify the Massena / Branson combination to include details as disclosed by Sotomayor regarding more specific details of page generation to create mark up language documents, including formatting elements. All are well known in the art, because the mark up languages and formatting techniques produce pleasing, meaningful, documents that may be widely distributed over the network.

***Response to Arguments***

10. Applicant's arguments with respect to claims 1-20 are moot in view of all claims being canceled.

***Conclusion***

11. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

12. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mary Steelman, whose telephone number is (703) 305-4564. The examiner can normally be reached Monday through Thursday, from 7:00 A.M. to 5:30 P.M. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on (703) 305-4552.

The fax phone number is (703) 872-9306 for regular communications and for After Final communications. Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Mary Steelman



05/27/2004

  
TUAN DAM  
SUPERVISORY PATENT EXAMINER